PepSlide® Analyzer 1.5

User's Guide

04 November 2013 - Rev 7

PepSlide® Analyzer is only intended for research and not intended or approved for diagnosis of disease in humans or animals.

Copyright 2009-2013 SICASYS Software GmbH. All Rights Reserved.

SICASYS Software GmbH
Im Neuenheimer Feld 583
D-69120 Heidelberg
Germany

Phone +49 (62 21) 7 28 50 40
Fax +49 (62 21) 7 28 48 94
Email info@sicasys.de
Web www.sicasys.de

Table of contents

1 Introduction 1

1.1 Installation 1

1.2 Product Activation 2

1.3 Concepts 2

1.4 Peptide Array Data Analysis 3

2 Preparation of Array Data 4

2.1 Loading Data 4

2.2 Editing Images 5

2.3 Rotating Images and Arrays 6

2.4 Aligning Arrays 6

2.5 Adjusting Spot and Grid Size 7

3 Quantification of Array Data 8

3.1 Quantifying Array Data 8

3.2 Change of Intensity Values 9

3.3 Spot Detection Methods 10

3.4 Background Correction 10

4 Analysis of Peptide Libraries 13

4.1 Overlapping Peptide Library Analysis 13

4.2 Random Peptide Library Analysis 15

4.3 Substitution Peptide Library Analysis 15

4.4 Peptide Library Analysis with GAL Files 17

5 Statistical Data Mining Tools 19

5.1 The Input Dataset 20

5.2 Principal Component Analysis 20

5.3 Hierarchical Clustering 22

5.4 K-Means Clustering 23

6 Utilities 24

6.1 Search for Antigens and Proteins 24

7 Product Activation 26

8 Sample Datasets 28

9 Customer License Agreement 30

10 Index 32

1 Introduction

PepSlide® Analyzer (PSA) supports microarray data analysis and is particularly suited for peptide arrays and peptide microarrays. In addition, the software provides statistical data mining tools for discovering parameters influencing the study and their relation.

Furthermore, PSA facilitates analysis of peptide libraries. The software can refer the quantified data to the peptide library design and the intended application, thus it is able to provide meaningful information with respect to the application. If e.g. a target protein or antigen is translated into overlapping peptides for epitope mapping, PSA can suggest a grouped list of peptides which represent an epitope or consensus motif of an antibody or serum sample.

PSA is part of the **PepSlide®** software suite. The other program is **PepSlide® Designer (PSD)**. PSD facilitates the design of peptide arrays and peptide microarrays, providing tools for interactive peptide library design and graphical array layout.

1.1 Installation

PSA is natively supported on Windows and Mac OS X platforms, thus the software is able to run with high performance and native look. Installation of PSA requires rights of a system administrator.

Hardware Requirement

Minimum hardware: 1.5 GHz Processor, 1GB RAM.

Recommended hardware: 2.0 GHz Dual-Core or faster Processor, 2 GB or more RAM.

Windows Platforms

PSA works on Windows XP, Windows 7, and Windows 8. Simply run the PSA setup. If the current Windows account is not an administrator, you will be asked to input an administrative account and its password.

Mac OS X platforms

The software runs on Mac OS X 10.7 and 10.8. Unzip the package and double-click on the .pkg file to launch the installer. During the installation you will be prompted to provide a system administrator's account and password. Upon completion, PSA is installed in the */Applications/PepSlide/Analyzer* folder.

1.2 Product Activation

After installing PSA on Windows, you need to *activate* PSA with a *trial serial number* obtained from the software provider or its distributors. This enables the use of PSA with full functionality for *30 days for free*. Please refer to the product activation steps in Section 6. The trial time for PSA on Mac OS X platforms is handled automatically and does not require this step.

When the free trial time has expired, you can continue using PSA by purchasing a *perpetual license* or a *term license* from the software provider or its distributors. Upon the purchase, you receive a *serial number* and use it to *activate* the license. These steps are described in Section 6.

1.3 Concepts

In the remaining of this manual, the term *array* is used to mean both *array* and *microarray*, unless otherwise stated.

The diagram of a typical peptide array analyzed by PepSlide® is outlined in Figure 1.

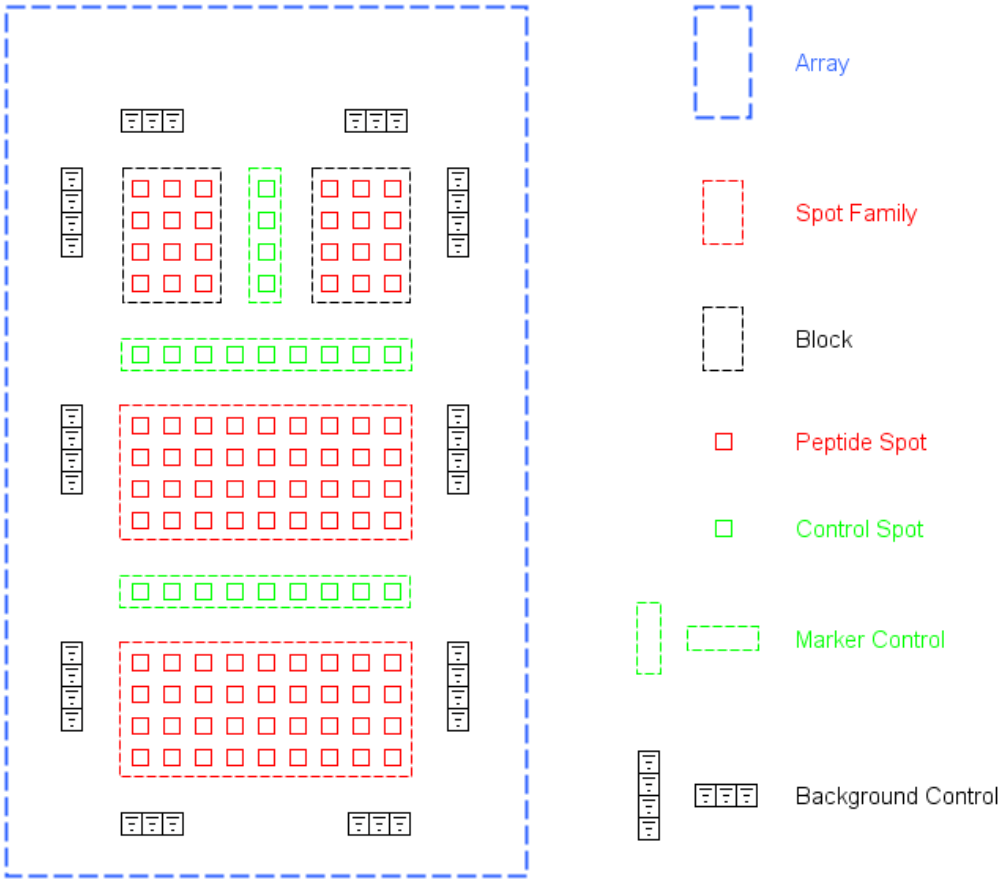


Figure 1: Peptide Array Diagram.

A s*pot* represents a single peptide spot located in one cell on the array. It can also be a control spot made of e.g. a known epitope. In the diagram, the spots are drawn as squares for the explanation purpose. Depending on the peptide array synthesizer, they may have rectangular or circular shape, as images of some actual spots shown in Figure 2.

A *spot family* (*SF*) consists of spots generated from the same source. For peptide arrays an SF can be a *peptide library* or a control. A *block* is a group of spots located next to each other, just like its definition in GAL files. The distinction between a block (contains only spots and their annotation) and an SF (a peptide library and may consist of multiple blocks) enables a convenient use of GAL files.

A *marker control* is a group of control spots arranged on either the same row or the same column of the array. It is used for the purpose of marking or orientation. A *background control* is used for the background correction during the data quantification. Its elements are *virtual* spots, i.e. neither peptide spot nor control spots, being used solely for defining the image regions from which the signal for background values is obtained.

1.4 Peptide Array Data Analysis

A PSA analysis project, including the analysis results and the paths to the input data, can be saved to a PepSlide® Analyzer file (*.psa) and reopened with the *Project* menu. Typical tasks of peptide array data analysis include:

1. Quantification of array data
    - Load the scanned images and the array.
    - Align the array to the images.
    - Quantify the array data.
2. Analysis of peptide libraries to select peptides of interest.
3. Discovery of parameters influencing the study and their relation with data mining tools.

The following sections explain how to do these tasks with PSA.

2 Preparation of Array Data

2.1 Loading Data

Two input data are required for analyzing a peptide array:

- Scanned images of the array in the TIFF format.
- The array design: Either a GAL file (*.gal) or a PepSlide® Designer file (*.psf).

PSA supports 8-bit or 16-bit grayscale images or 24-bit color images. For the best image quality *16-bit grayscale TIFFs* are recommended. Please note that compressed images are not supported. Therefore, please disable image compression when saving the scanned images with your microarray scanner software. If you are using GenePix Pro software, uncheck the *Use TIFF LZW compression (lossless)* option in the *Save Images* dialog.

The input data is loaded as follows:

- Images: Choose the *Images* > *Open Image* menu. For grayscale images, select either the *Red* channel or the *Green* channel to display the signal of each image.
- Array design: Choose the *Arrays* > *Open Array* menu.

Image editing and data quantification tasks are handled in the *Imaging* sheet. After being loaded, the images and the array are shown in the graphical canvas on the right side of this sheet. To obtain an appropriate view, you can use the *Zoom In* and *Zoom Out* buttons on the toolbar or select a predefined zoom level in the *Zoom* combo-box. Alternatively, an arbitrary value can be entered directly into this combo-box. A control panel is located on the left side of the *Imaging* sheet. It shows properties of the image and selected objects in the array, in addition to options for image processing and data quantification tasks.

If the array file is of PSF format, spots are grouped into *Spot Families (SF)*. To view the information about an SF, first activate the *Spot Family* tab in the *Design Information* section of the *Imaging* sheet, and then click on the SF in the canvas.

In case the array file is of GAL format, the software shows properties of a GAL block in the *Block* tab, which is located next to the *Spot Family* tab. When the *Block* tab is activated, you can select a block, align it by moving and rotating, and quantify its spots.

The information about a peptide spot, such as its unique position on the array and the sequence of amino acids, can be viewed similarly: First activate the *Spot* tab and then click on that spot. Within each spot, the *spotted region* is bounded by a *dashed rectangle* or *circle* depending on peptide array synthesizer (Figure 2). During the array alignment, you will reallocate the array such that the spotted regions as close to the spots in the images as possible.

(a) Rectangular Spot (b) Circular Spot

Figure 2: Spot Images in PepSlide® Analyzer.

2.2 Editing Images

Aligning arrays can be difficult if the spot visibility in the image is poor. This may be due to low signal intensities or noise of the image. In the *Imaging* sheet > *Image Processing* section > *Pixels* tab, you can find utilities to improve the spot visibility.

Adjustment of Brightness and Contrast

Adjusting brightness and contrast can make the spots more visible for the array alignment. When the *Contrast* check-box is selected and the *Automatic Enhancement* option is chosen, PSA examines the image data and adjust the pixel intensity values such that spots are as visible as possible.

Brightness and contrast can also be manually adjusted, either by setting an *arbitrary* valid value *directly* in the combo-box or using the up/down control to change one unit. The valid range of brightness and contrast values for manual adjustment is [-99, 99].

Noise Filtering

Noise in the loaded image(s) can be reduced by selecting the *Noise Filtering* check-box.

Inversing

An array image can be converted to a negative image by selecting the *Inversion* check-box.

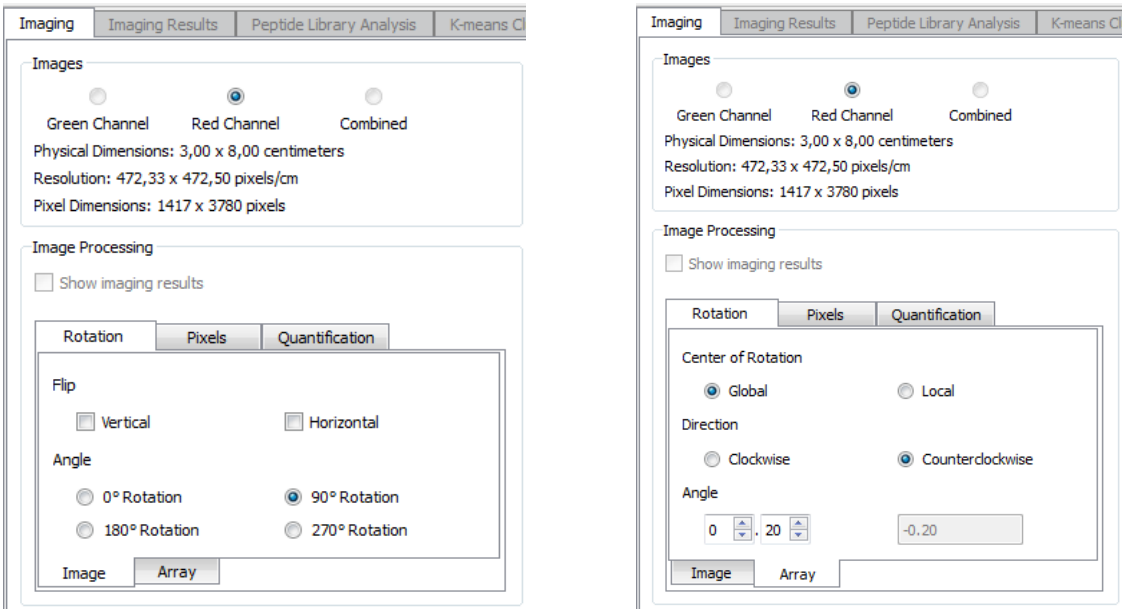
2.3 Rotating Images and Arrays

PSA supports a comprehensive set of tools for rotating images and arrays. These tools are located in the *Imaging* sheet > *Image Processing* section > *Rotation* tab.

To rotate the images, select the *Image* tab to display options as shown in Figure 3(a). One can flip the images and/or rotate them at angles of 90°, 180°, or 270°.

If the array slightly deviates from the images at a small angle, it is recommended that the array (i.e. related SFs or blocks) be rotated, since image rotation may change the image data. To do that, select the *Array* tab to display parameters as shown in Figure 3(b). Selected SFs or blocks can then be rotated at an *arbitrary angle* in clockwise or counterclockwise direction, with a rotation center:

- *Global*: the top-left of the image.
- *Local*: the top-left of each SF and block.



(a) Rotating an image (b) Rotating an array

Figure 3: Rotation of Images and Arrays.

2.4 Aligning Arrays

Before aligning the array, you may need to adjust the images' contrast and brightness to make the spots more visible. In addition, the rotation of the images or the array may be necessary.

If a PSF array file is loaded, first select the spot family (SF) containing the spots and then adjust the SF's position so that the spotted regions match their counterpart on the image. Please note that one

cannot move or rotate spots; instead this is done by means of the SF containing them. Therefore, in the *Imaging* sheet > *Design Information* section, the *Spot Family* tab must be activated.

- Selecting SFs
  - Use Ctrl-A to select all SFs in the array.
  - To select individual SFs, click on the respective SF in the canvas. Hold down the Ctrl key for selecting multiple SFs simultaneously.
- Moving SFs: Click on the selection and drag it to the desired position.

When a GAL file is loaded, blocks can be selected, moved, and rotated like SFs. Please remember to activate the *Block* tab before using functions on blocks.

The aligned array file can be saved with Ctrl-S or *the Arrays* > *Save* menu. It can be saved to another array file using the *Arrays* > *Save Array As* menu.

2.5 Adjusting Spot and Grid Size

In addition to reallocating SFs or blocks, you may need to adjust the spots' diameter or the distance between them, so that the spots and their spotted regions in the array fit their counterpart in the image(s).

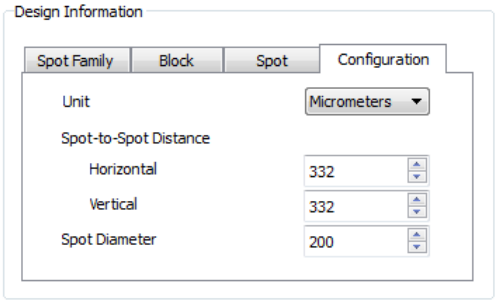


Figure 4: Circular Spot Size.

These parameters can be changed in the *Imaging* sheet > *Design Information* section > *Configuration* tab. The software reads the spot's specification from the array file and displays the corresponding settings.

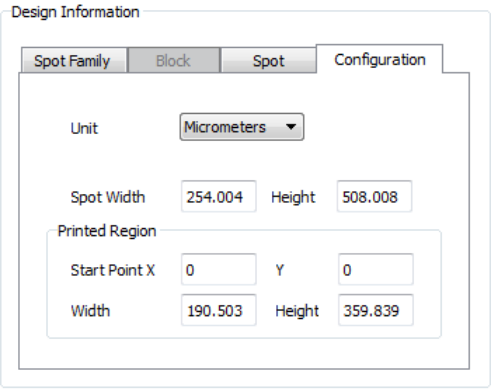


Figure 5: Rectangular Spot Size.

Figure 4 shows the spot diameter and the distance between them for circular spots. Similarly, Figure 5 displays the settings for rectangular spots. Saving the array file commits changes made to these parameters permanently.

3 Quantification of Array Data

3.1 Quantifying Array Data

You need to activate the *Imaging* sheet to perform the array data quantification.

- Right-click at a point within the image but outside any SF or block. Choose *Image Processing (entire array)* to quantify all spots of the array.
- To quantify selected SFs or blocks, right-click on the selection and choose *Image Processing*.

For each spot in the array, the median and the mean of its *raw*, *background*, and *foreground* values in each channel (Red and/or Green) are calculated. *Raw* represents the intensity value of the spot's signal. *Background* is the estimated value of the signal caused by non-specific binding. The value of interest is *foreground*; it is computed by subtracting the background from the raw value. If the raw value of a spot is smaller than its background value, the spot is flagged *Error* and its foreground value is set to zero.

In the *Imaging* sheet, you can view the spots and their quantified data simultaneously (Figure 6). The image region corresponding to the selected spot and its neighbors are displayed in the *Spot Image* widget. In addition, selecting a spot in the array highlights its quantified data in the table and vice versa. Please remember that in the *Imaging* sheet > *Design Information* section, the *Spot* tab must be activated for browsing spots and their data.

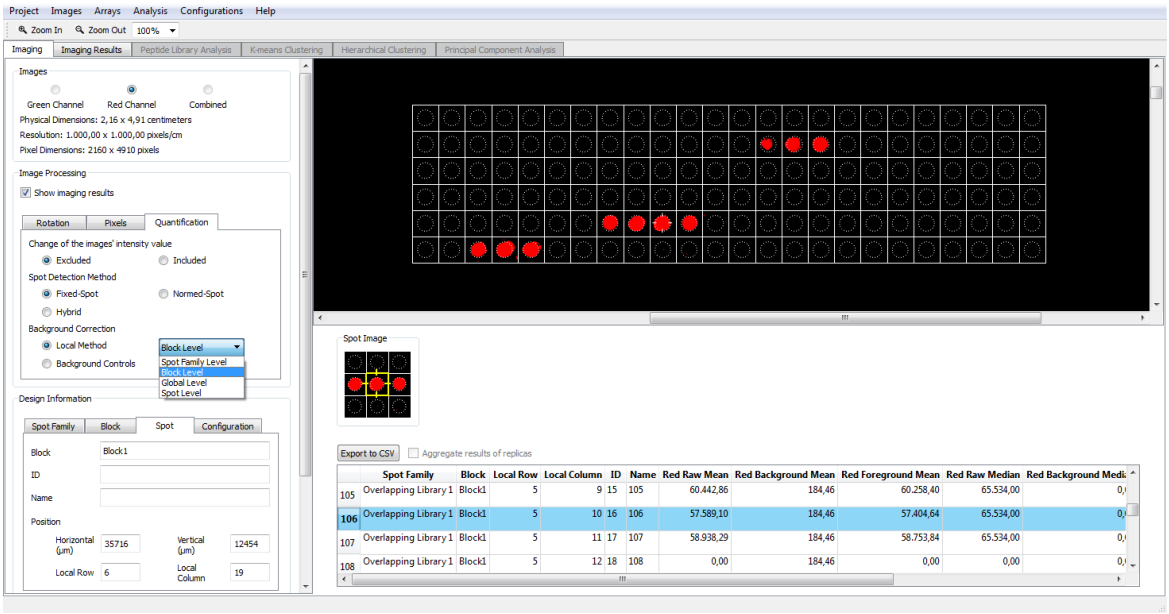


Figure 6: Quantified Data.

You can also browse the spots' quantified data in the *Imaging Results* sheet, which shows only the table of quantified data and the *Spot Image* widget. In both *Imaging* and *Imaging Results* sheets, you can export the quantified data to a CSV file for further analysis.

Aggregating results of replicas

When a peptide is replicated the software also provides the intensity value of the peptide calculated by aggregating signals from its replicated spots. If e.g. P1 and P2 are the replicas of a peptide P, then the quantified results include the intensity values for P, in addition to those for P1 and P2. Taking the raw median value of P for example, it is the median value of signals from both P1 and P2 spots, which cannot be calculated based on the median values of P1 and P2.

This provides an additional view to the replicated data. To control whether the aggregated data is calculated and shown, use the *Aggregate results of replicas* check-box.

Customizing the Quantified Data

Click the *Configurations* menu and choose *Imaging Result Values* to show the setup dialog. Here, you can add or remove values to be calculated during the data quantification.

3.2 Change of Intensity Values

During the array alignment, you may have adjusted the contrast and/or the brightness of the images to make the spots visible. This changes the images' signal and alters the analysis results. It is recommended that *the original image data be used for the quantification.* Therefore, by default *the quantification procedure uses the original image's signal*, i.e. it *excludes* changes made to the image's signal due to inversion, noise filtering, or the adjustment of contrast and/or brightness. This option can be intentionally changed as follows. In the *Imaging* sheet > *Image Processing* section > *Quantification* tab, select the *Included option* below the *Change of the images' intensity value* label.

To use the original image data for the data quantitation again, select the *Excluded option.* Please note that when the this option is chosen, the *Spot Image* widgets in the *Imaging*, *Imaging Results*, and *Peptide Library Analysis* sheets show the spot's counterpart image using the original image's signal, even though the image's contrast or brightness has been changed.

3.3 Spot Detection Methods

In the *Imaging* sheet > *Image Processing* section > *Quantification* tab, the *Spot Detection Method* defines how the raw and background values of a spot are calculated.

- *Fixed-Spot*: The software uses the signals in the *spotted region* to compute the raw value. The background value is calculated based on the signals in the remaining region within the spot. This is the default method. It is recommended if the spots in the images look consistent and match well with the spotted regions in the array.
- *Other methods*: The software combines the known information about the spotted region with the detected signals within each spot. It then *flexibly* separates the region consisting of signals for the raw value from the one for the background value.

The background correction mechanism described above is called the *Local Method*; it uses only the *background signals*, i.e. signals within the spots whose intensity values are used for calculating the background values. By means of background controls, you can have background values using the signals from any region of the image.

3.4 Background Correction

Background correction methods can be selected in the *Imaging* sheet > *Image Processing* section > *Quantification* tab. By default, the *Local Method* is employed.

Local Method

PSA supports different levels of background correction with local methods (Figure 7). The default one for PSF array files is the *Spot Family Level*. In the case of GAL array files, it is the *Block Level*.

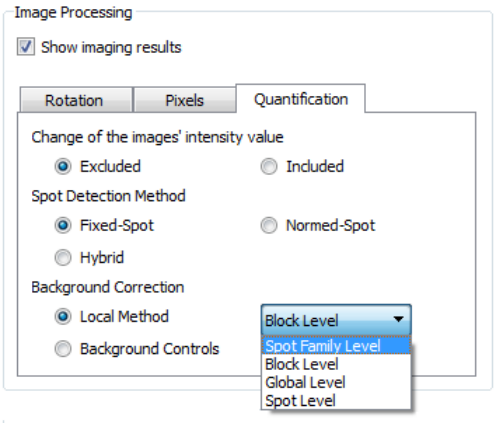


Figure 7: Background Correction Level

With the *Spot Family Level*, all spots in a spot family (SF) have the same background value. It is computed as follows. First, a list of background signals of all spots in the SF is created. The mean and median of the signal intensity values in this list are then used as the background value.

The *Block Level* is available only for GAL files. A single background value is applied to all spots in a block, using similar calculation as at the Spot Family Level.

You can choose to have a global background value for the entire array with the *Global Level*. The calculation is based on the background signals of all spots in the array.

At the *Spot Level*, only *the* spot's background signals are used to compute its background values. Therefore, it is likely that the background values are different between spots.

Background Controls

Instead of having the background values locally calculated, you can use background controls to explicitly specify image regions from which the background values are calculated. One may choose to have a *global background value for the entire array* using a background control as follows.

- Right-click at a point in the image where the signal may represent the background for the spots. Choose *Create Background Control*. Specify the shape and the number of spots.
- Perform the data quantification. All spots in the array now have the same background value which is obtained from the background control. We call that an "*association*" between the SFs and the background control.

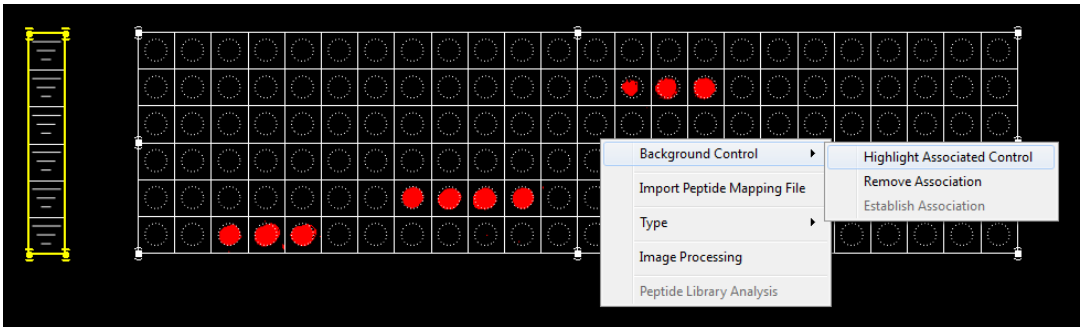


Figure 8: Associated Background Control.

In addition to a global background value for the entire array, one can flexibly choose *an individual background value for a spot family*. This is done by creating a background control just for the SF and establishing the association between them.

- Right-click on the SF to popup the context menu. Choose *Background Control* > *Remove Association* to release the SF from the first background control.
- Right-click on the SF again and choose *Background Control* > *Establish Association*. Click on the background control with which you want to associate this SF. You can check with which control an SF is associated using the *Highlight Associated Control* context-menu (Figure 8).

For GAL files, having a global background value or an individual background value for a *block* can be done similarly; you create a background control for the blocks and associate them.

Please note that the background correction method is automatically set to *Background Controls* after the creation of the first background control. In the *Imaging* sheet > *Image Processing* section > *Quantification* tab, you can switch back to use the *Local Method*.

4 Analysis of Peptide Libraries

After having the peptide array data quantified, you may want to examine it and select peptides of interest with respect to the peptide array application. E.g. if an antigen is translated into overlapping peptides for epitope mapping, you may be interested in a list of peptides which represent an epitope. This can be done conveniently with PSA's peptide library analysis tools. By referencing the quantified data to the peptide library design, which keeps the antigen's amino acid sequence and the overlapping step, PSA is able to locate the peptides representing the epitope.

The software supports analysis of peptide library of the following types:

- *Overlapping library*: Overlapping peptides translated from a protein.
- *Random library*: Peptides generated stochastically.
- *Substitution* (also called *permutation* or *positional scanning*) *library*: Variants of a known peptide created by replacing its residues with other amino acids.

As mentioned, PSA represents a peptide library as a spot family (SF). If the array file is of PSF format, simply select the relevant spot families, right-click on the selection, and choose *Peptide Library Analysis*. The results are shown in the *Peptide Library Analysis* sheet.

Analysis of a peptide library cannot be performed with GAL files in the first place. However, you can easily specify the peptide library design for related GAL blocks. The analysis can then be preceded as with SFs. These steps are detailed in Section 4.4.

4.1 Overlapping Peptide Library Analysis

This analysis aims at selecting peptide spots with strong signals. It is also able to detect a consensus motif shared by overlapping peptides.

A two-dimensional chart in the center of the sheet depicts the peptide spots (Figure 9). Their coordinates can be selected among the quantified intensity values. By moving the lower and upper threshold bars, the chart facilitates the selection of spots whose intensity values being within the two thresholds. Selected spots are then populated into the table below the chart.

By clicking on a spot in the chart or selecting the corresponding row in the table, you can view the spot information such as the peptide and its counterpart image. The information is shown in the *Spot Details* widget located on the right of the sheet. The chart, the table, and the *Spot Details* widget thus provide different angles to the data of the peptide library.

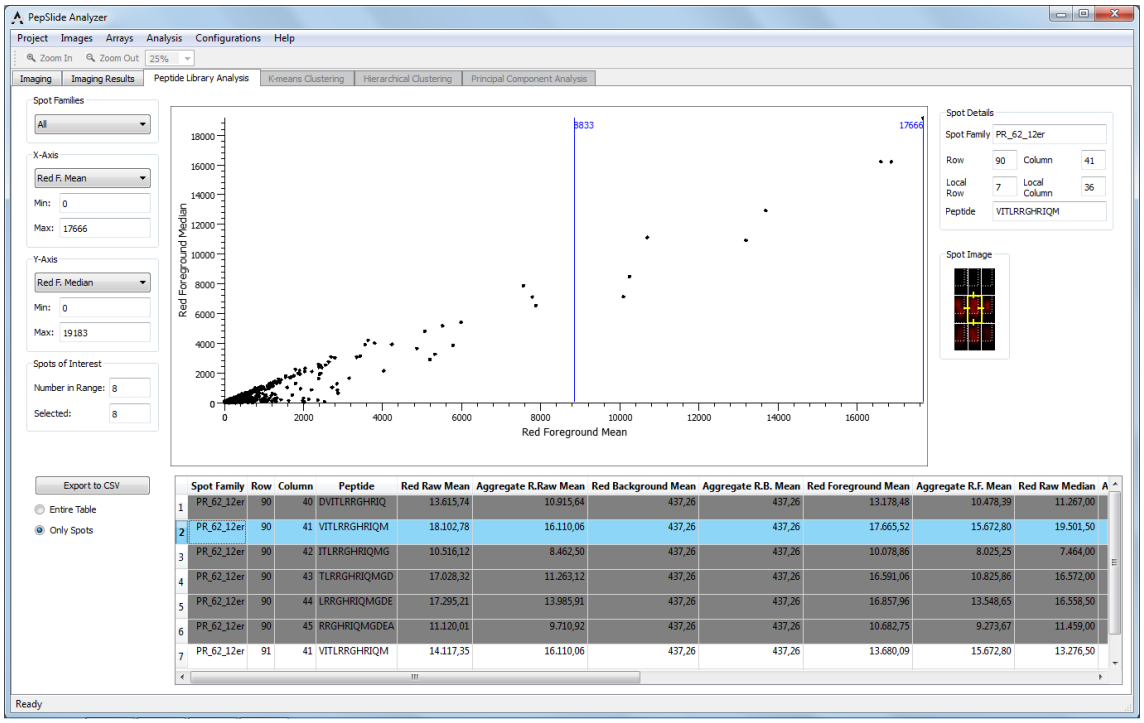
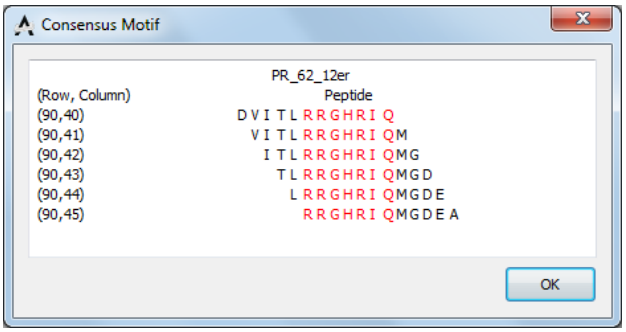


Figure 9: Overlapping Peptide Library Analysis.

You can export the data in the table to a CSV file. The chart can also be saved to an image file.

Identifying Consensus Motifs

In the table below the chart, overlapping peptides sharing a consensus motif are highlighted with an individual color. In Figure 9, rows corresponding to these peptides are in grey. Among them, the one being selected for review is highlighted in blue; its information is shown in the *Spot Details* widget.



Right-click on the highlight and choose *Show Consensus Motif* to view the information (Figure 10). Data in this dialog can be copied and pasted into other software, e.g. a word processing program.

Figure 10: Consensus Motif.

In the sample data package, the *overlapping1* folder contains a test dataset for this analysis.

4.2 Random Peptide Library Analysis

This analysis facilitates browsing and selecting peptide spots with strong signals for further optimization. Except for the detection of consensus motifs, the user interface and functions of this analysis (Figure 11) are the same as those in the analysis for overlapping peptide libraries.

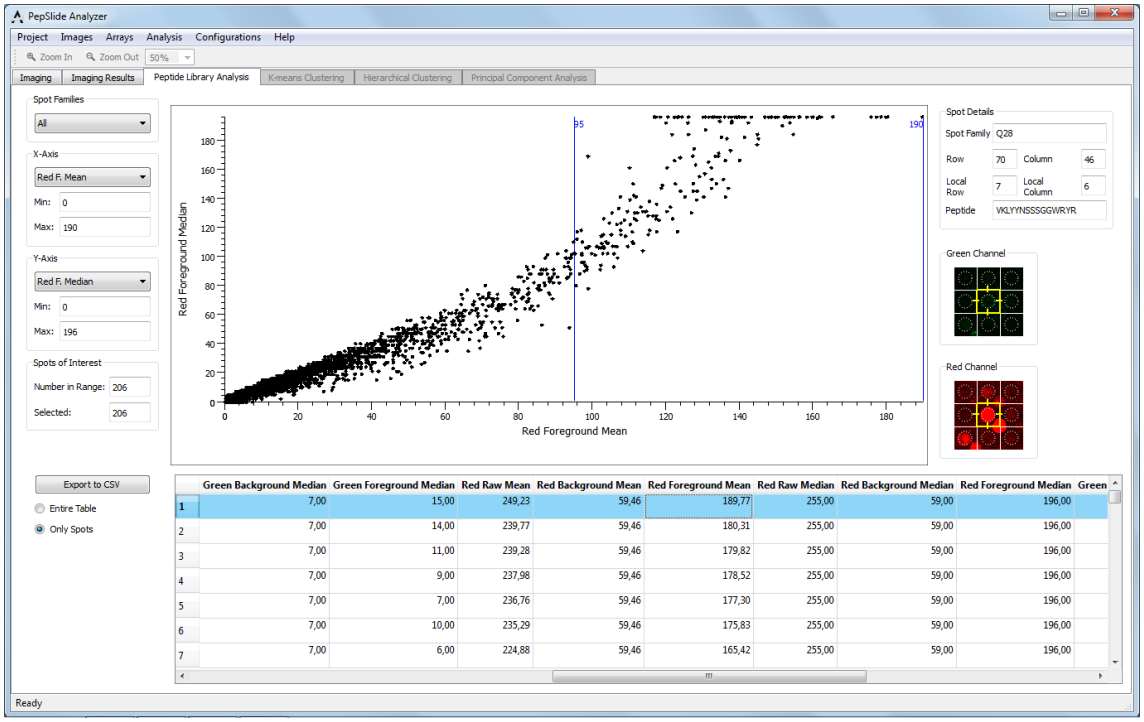


Figure 11: Random Peptide Library Analysis.

In the sample data package, the *random1* folder contains a test dataset for this analysis.

4.3 Substitution Peptide Library Analysis

Peptides of a *substitution* peptide library are created by replacing residues of a known peptide, the so-called *parent peptide*, with some other amino acids. The purpose of this analysis is thus to examine all variants of the parent peptide and select those of your interest. Figure 12 displays the analysis results of a substitution peptide library whose parent peptide is *NYGKYE*.

The variants of the parent peptide and their intensity value are simultaneously shown in a three-dimensional chart. Here, the (P)-Axis lists residues of the parent peptide and the (S)-Axis contains amino acids for replacing, thus a “point” in the (PS)-plane features a variant. The intensity value of a variant is represented by the height of the corresponding bar along the Z-Axis. You can choose to evaluate another quantified value in the *Z-Axis* section in the left panel. The perspective of the chart can be changed by holding-down the left-button of the mouse and moving it.

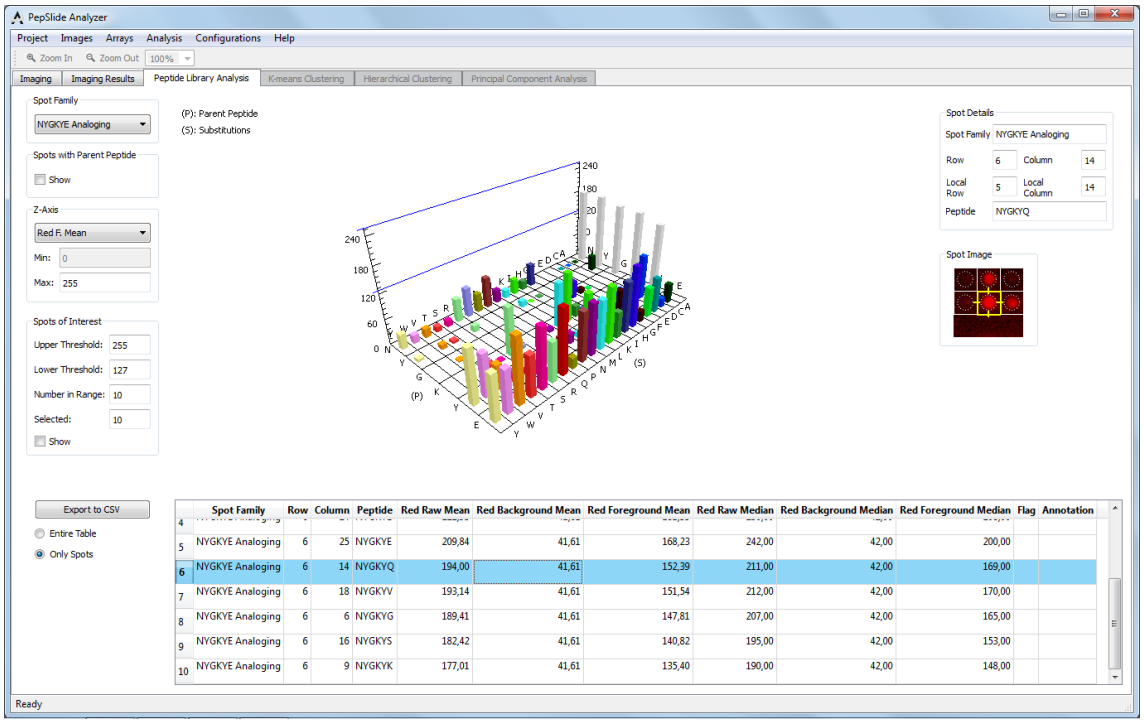


Figure 12: Substitution Peptide Library Analysis.

Spots are selected by entering the lower and/or upper threshold values in the *Spots of Interest* section in the left panel. The selection will be populated into the table below the chart. Similar to other analyses, you can view information about a spot in the *Spot Details* widget. Selecting a spot in the table highlights its version in the chart and vice versa.

The chart can be saved to an image file. In addition, the data in the chart and the table can be exported to a CSV file.

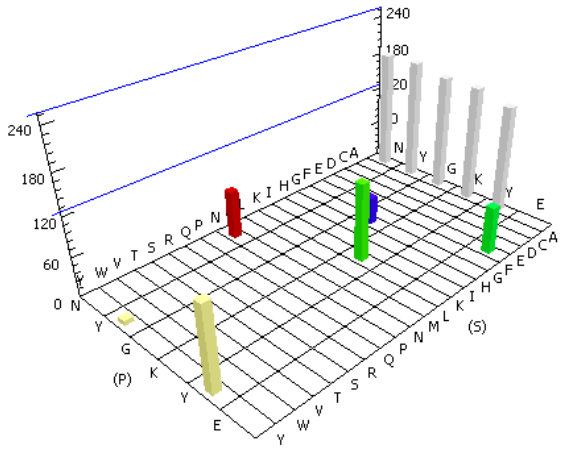


Figure 13: Parent Peptide Spots.

Quality Control with Parent Peptide Spots

You can choose to display only spots having the same amino acid sequence as those of the parent peptide. Since these spots are the same peptide, their intensity values should not be considerably different (Figure 13). This option is found in the *Spots with Parent Peptide* section in the left panel.

In the sample data package, the *substitution1* folder contains a test dataset for this analysis.

4.4 Peptide Library Analysis with GAL Files

You can conveniently use array files of GAL format, even for the analysis of peptide library. Whether the peptides of a *peptide library (PL)* are spotted into one block or different blocks, you can easily reconstruct the PL and then analyze it as explained in the previous sections. The reconstruction can be done in two steps:

- Specify a peptide library design for the blocks. This creates a PL consisting of these blocks.
- If the GAL file does not contain the peptide in the spots' ID, import the peptide mapping file.

Each step is described in detail below. For reading convenience, we assume that an overlapping PL is being reconstructed. The procedure is the same for substitution and random peptide libraries.

Specify a Peptide Library Design for GAL Blocks

- Hold down the Ctrl key while clicking on the blocks to select them simultaneously.
- Right-click on the selection and choose *Type* > *Overlapping Peptide Library*.
- Enter the design of the peptide library (Figure 14) including the protein, the shift step, and the peptide length.

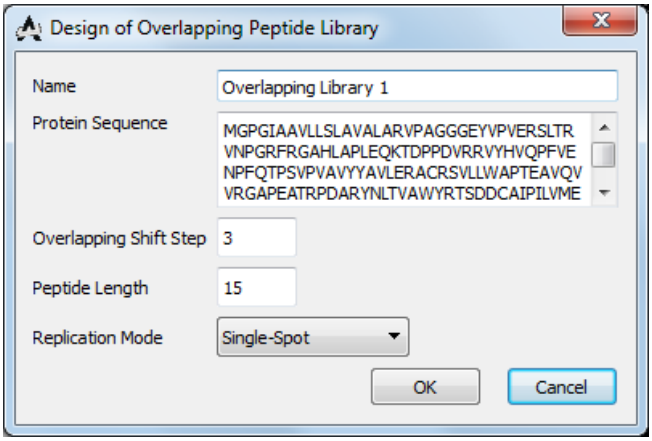


Figure 14: Overlapping Peptide Library Design.

An overlapping peptide library consisting of these blocks, represented by PSA as a spot family, has been created. You can activate the *Spot Family* tab and review the information of the PL.

Import the Peptide Mapping File

Please note that this step is only necessary if the spots' ID field in the GAL file does not contain the peptide yet, as illustrated in Figure 17(a).

- Activate the *Spot Family* tab to work on peptide libraries.
- In the canvas, select the PL by clicking on one of its blocks. Right-click on the selection and choose *Import Peptide Mapping File* from the context menu (Figure 15).
- Specify the linker which will be excluded from the peptides. Leave it blank if no linker is used.
- Select the file which maps the peptides of the PL to their spot ID in the blocks.

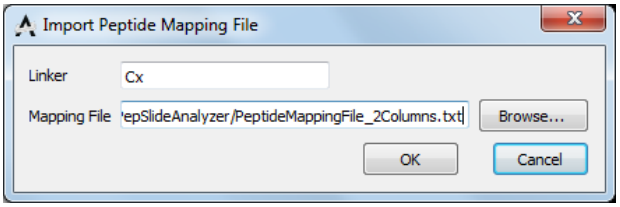
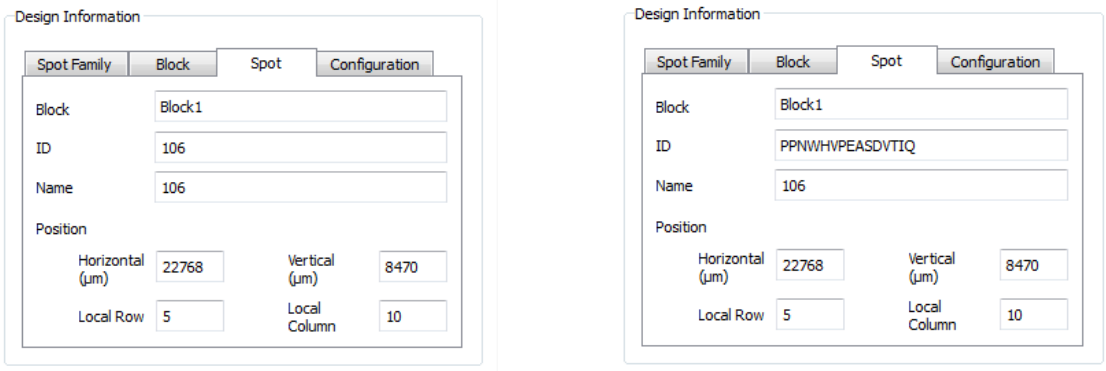


Figure 15: Import the Mapping File.

Spot_ID	Peptide
1	CxMGPGIAAVLLSLAVA
2	CxGIAAVLLSLAVALAR
3	CxAVLLSLAVALARVPA
...	
106	**CxPPNWHVPEASDVTIQ**
...	

Figure 16: The Peptide Mapping File.

The peptide mapping file is a Tab-delimited text file or CSV file. It has two columns; the first one contains the ID of the peptides in the blocks and the second column contains the peptides. The first line of the file is reserved for the column headers and will be skipped. An example of the file's content is shown in Figure 16. In this example, the linker is Cx.



(a) Spot ID from the Original GAL File (b) Spot ID with the Peptide

Figure 17: Assigning the Peptide to a Spot.

After importing the peptide mapping file, activate the *Spot* tab in the *Design Information* section. Now, you can view the peptide in the *ID* field when clicking on a spot in the canvas, as illustrated in Figure 17(b). The overlapping peptide library has been completely reconstructed and can be analyzed as described in Sections 4.1 to 4.3. Please note that *saving the GAL file also keeps the information about the peptide library design*.

In the sample data package, the *overlapping3* folder contains a test dataset for this function.

5 Statistical Data Mining Tools

Statistical data mining tools assist you to discover useful information from complex datasets. For example, in this study[1] the researchers were able to distinguish the patients before and after the medical treatment with the assistance of Principal Component Analysis and Hierarchical Clustering Analysis. These statistical analysis tools are available in PepSlide® Analyzer. You can use them in the following applications, including but not limited to:

Statistical Analysis Tool	Possible Applications
Principal Component Analysis (PCA)	• Find features (peptides) and samples that most likely determine the variance of the data. • Search for views that are likely to separate the data and show the variance direction. • Reduce the dimension of the dataset and identify new influential variables.
Hierarchical Clustering Analysis (HC)	• Find possible association or relation between features (peptides) and/or samples. • Identify similarity between features or samples.
K-Means Clustering Analysis	• Classify data with similarity or association into groups.

By combining the above statistical analyses, you have powerful tools for the discovery of influential factors of your data and possible relation between them, e.g. between the peptides and/or the samples.

Notes on data source: The data from this work[2] is used to illustrate the analysis results in this section.

[1] Le Moyec, et al. (2005): Serum ^{1}H-nuclear magnetic spectroscopy followed by principal component analysis and hierarchical cluster analysis to demonstrate effects of statins on hyperlipidemic patients. NMR Biomed., 18: 421–429.

[2] Schröder et al. (2010): Dual-color proteomic profiling of complex samples with a microarray of 810 cancer-related antibodies, Molecular & Cellular Proteomics 9(6): 1271.

5.1 The Input Dataset

This section explains how to prepare and load the dataset for the Principal Component Analysis and the Hierarchical Clustering Analysis. An input dataset can be:

- compiled from a list of PepSlide® Analyzer files (*.psa);
- compiled from a list of GenePix Result files (*.gpr);
- a CSV file exported from a dataset.

The PSA and GPR files must contain the quantified array data for the same set of *features* (for peptide arrays a feature is a *peptide*). In other words, they all based on a *template array file*. In the loaded dataset, the name of a PSA or a GPR file is used as the name of a *sample*. The features' value is selected among the list of quantified intensity values, e.g. Red Foreground Mean, in the *Evaluation Parameter* section of the analysis sheet.

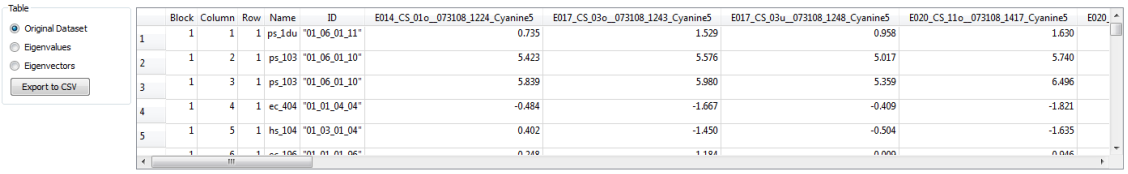


Figure 18: The Input Dataset

After being loaded, the dataset is shown in the table at the bottom of the analysis sheet (Figure 18). In the case the dataset is compiled from PSA or GPR files, the loaded data contains only the selected quantified values and thus has small size. Therefore, if you plan to analyze the dataset again, export it to a CSV file. It is much faster to load this CSV file than the set of PSA or GPR files in each analysis.

5.2 Principal Component Analysis

Click the *Analysis* menu and choose *Principal Component Analysis* to activate the PCA sheet. If you want to examine the effect of features (peptides) in your study, select the *Samples* option in the *Components* section. This *projects*, or *simplifies*, the dataset to the three principal samples. On the other hand, performing PCA with the *Features* option simplifies the dataset to the three principal features, which helps you to observe how the samples affect your study.

- Click the *Load Data* button to load the input dataset. Please refer to Section 5.1 for the preparation of the dataset.
- Specify the evaluation parameter which is one of the quantified values.
- Click the *Start PCA* button.

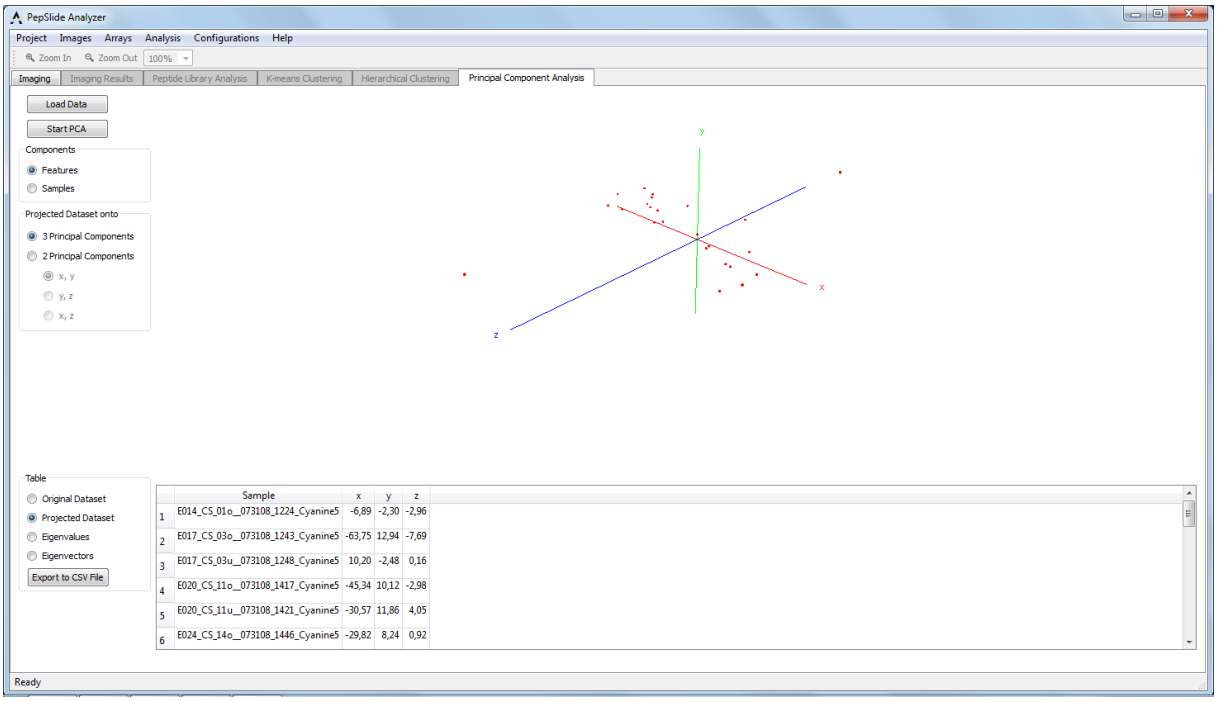


Figure 19: Principal Component Analysis.

The projection of the input dataset to the *three* principal features or samples is depicted in the PCA chart (Figure 19). You can change the view angle to the data by holding-down the left-button of the mouse and moving it. The chart is three-dimensional initially and can be switched to two-dimensional views (Figure 20). PCA charts can be saved to an image file using the *Export to Image* context-menu.

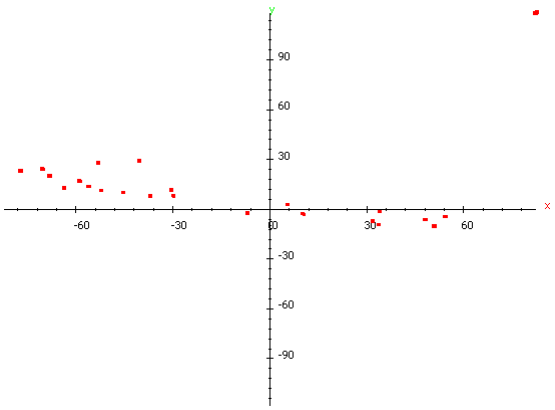


Figure 20: Projection on Two Principal Components

In the table, you can precisely examine the projected data by selecting the *Projected Dataset* option. Suppose that you performed PCA with the *Samples* option. You can assume that the projected dataset represents the results of a simplified study, which approximates the original study, with only three samples. The table contains the features (peptides) and their quantified values corresponding to three projected samples x, y, and z. You can then observe the effect of features (peptides). Similarly, the *Features* option leads to showing the samples and their projected values in the table. The projected dataset can be exported to a CSV file for further analysis.

In the sample data package, the *schroeder2010* folder contains a test dataset for this analysis.

5.3 Hierarchical Clustering

Click the *Analysis* menu and choose *Hierarchical Clustering* to activate the HC sheet.

- Click the *Load Data* button to load the input dataset. Please refer to Section 5.1 for the preparation of the dataset.
- Specify the evaluation parameter which is one of the quantified values.
- Choose to construct the clustering tree for features, or samples, or both.
- Select the desired *Distance Metric* and *Algorithm Type*.
- Click the *Start HC* button to begin analysis.

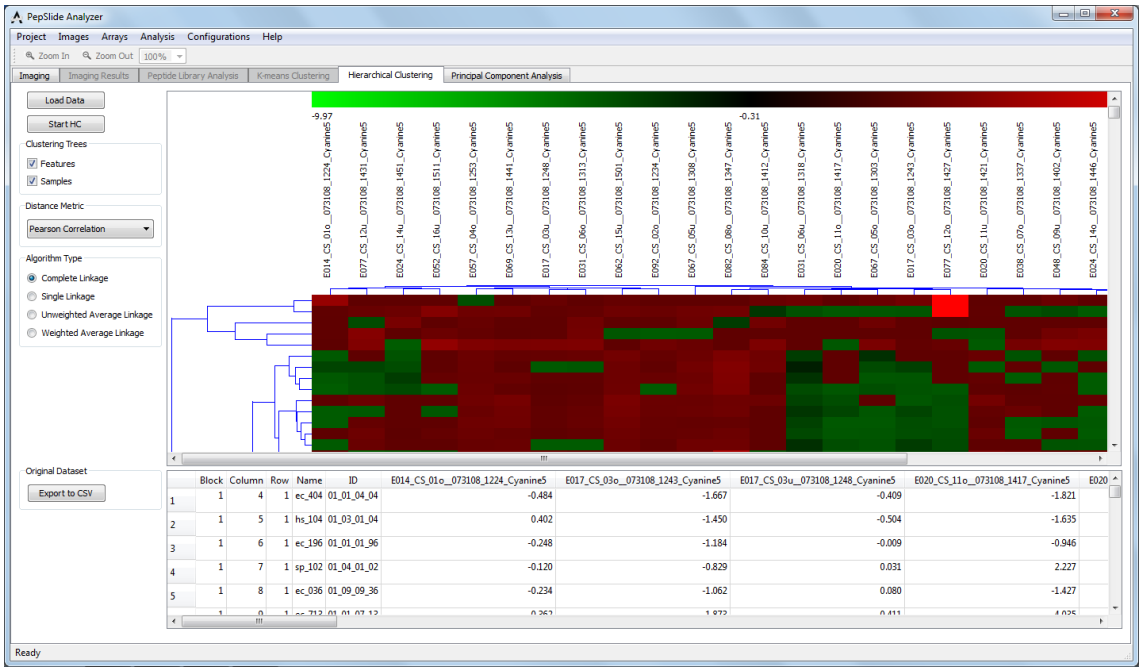


Figure 21: Hierarchical Clustering Analysis.

The clustering tree(s) are then constructed (Figure 21). Two features (peptides) considered being *related*, e.g. due to having similar effect in the study, are grouped into one *cluster*. Their *relationship* is represented by a line connecting them. A cluster might be related with a feature or another cluster. The relationship between samples and clusters of samples are represented similarly.

The values in the dataset, each represents the signal value of a feature in a sample, are graphically represented by means of a heat map. The clustering trees with the heat map can be saved to an image file using the *Export to Image* context-menu.

In the sample data package, the *schroeder2010* folder contains a test dataset for this analysis.

5.4 K-Means Clustering

After analyzing *overlapping* or *random* peptide libraries, you can use K-means Clustering to classify peptides into different groups based on their quantified intensity values.

- Click the *Analysis* menu and choose *K-Means Clustering* to activate the setup dialog.
- Enter the desired number of clusters, i.e. groups, and customize their color if necessary.
- Click OK: The chart then shows spots highlighted in their cluster's color (Figure 22).

In the *Clusters* section in the left panel, select the cluster of interest. The table then shows only spots of that cluster.

Data in the table can be exported to a CSV file. The chart can be saved to an image file.

Figure 22: K-Means Clustering Analysis.

In the sample data package, the dataset in the *random1* folder can be used to test this analysis.

6 Utilities

6.1 Search for Antigens and Proteins

It is helpful to know if a generated peptide is part of an antigen or a protein. PSA assists you to do that easily, for either the entire peptide array or only selected peptides. You first need to setup the list of antigens and proteins that might be related to you study. Then during quantification of peptide array data or analysis of peptide libraries, you can conveniently search for antigens or proteins that contain a peptide. The list can be easily imported and updated.

Setting up the List of Antigens and Proteins

Use the *Configurations* > *Antigen & Protein List* menu to launch the setup dialog (Figure 23). An antigen or a protein is stored in one row in the table. Its data is contained in five columns:

- Nr.: A numeric ordering number.
- ID: Unique identification of the item. It will be shown in the search result of a peptide if the antigen or protein contains the peptide.
- Name: The item name.
- Description: Further information about the item.
- Amino Acid Sequence: The item's sequence of amino acids.

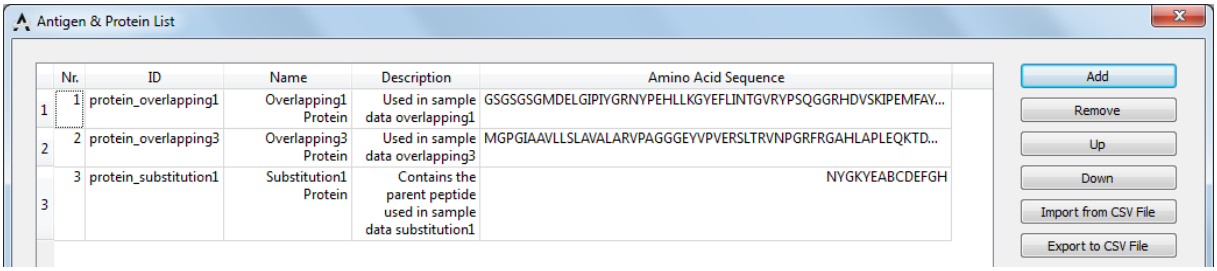


Figure 23: List of Antigens and Proteins.

If you already have a list of antigens and/or proteins contained in a spreadsheet file, you can export that list to a CSV file and then conveniently import that list into PSA. Alternatively, you can manually add an item to your list with the *Add* button. Use the *Remove* button to delete an existing item.

Figure 24 shows an example CSV file containing a list of proteins. The CSV file format can be either *Tab delimited* or *Semicolon delimited*. This option can be chosen during the export with your spreadsheet software. The first line of the file is the header and will not be processed. PSA will import items from the second line of the CSV file. If you import another list, the new items will be appended to the current list. Please note that the current version of PSA does not check if items' ID is duplicated. Therefore, please prepare the list with unique IDs.

	A	B	C	D	
1	No.	ID	Antigen/Protein Name	Description	Amino Acide Sequence
2	1	protein_overlapping1	Overlapping1 Protein	Used in sample data overlapping1	GSGSGSGMDELGIPIYGRNYPEHLLKGYEFLINTGVRYPSQGGRHDVSKIPEMFAYDI
3	2	protein_overlapping3	Overlapping3 Protein	Used in sample data overlapping3	MGPGIAAVLLSLAVALARVPAGGGEYVPVERSLTRVNPGRFRGAHLAPLEQKTDPP
4	3	protein_substitution1	Substitution1 Protein	Contains the parent peptide used in sample data substitution1	NYGKYEABCDEFGH

Figure 24: A List of Antigens and Proteins in the CSV File

You can export the current list to a CSV file for backup purpose or using that list on another computer. To do that, launch the setup dialog again and click the *Export to CSV File* button.

Please note that there are two example CSV files in the *antigenProteinList* folder in the sample data package. They contain the same list of proteins but are stored in two different formats. You can use them to test the function or as a template to create your own list.

Searching for Antigens & Proteins

In the *Imaging Results* sheet, you can perform the search for the entire array. Alternatively, in the *Peptide Library Analysis* sheet the search is only applied to selected peptides. In either cases, click on the *Find Antigen/Protein* button to start the search. The *Search for Antigen & Protein* dialog (Figure 25) determines how PSA carries out the search for a peptide:

- *Stop at the first finding*: The search is finished as soon as an antigen or a protein in the list is found containing the peptide.
- *Search for all matches*: The entire list will be scanned. The search returns the ID of *all* antigens or proteins that contain the peptide.

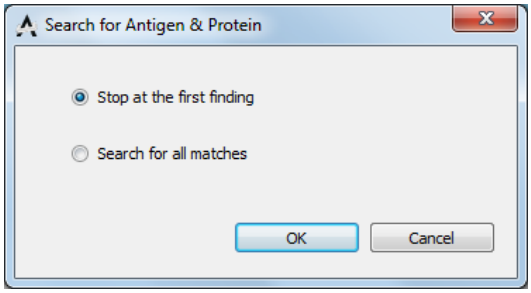
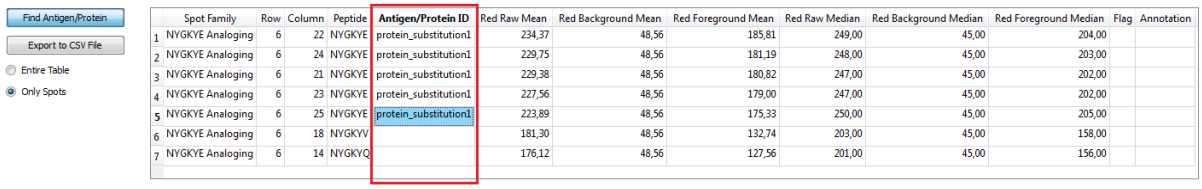


Figure 25: Search Options.

The search results are shown in the *Antigen/Protein ID* columns of the table of quantified data or the table of selected peptides. Figure 26 illustrates the latter case. You can export the table's data, including the peptide and the antigen/protein ID columns, to a CSV file.



	Spot Family	Row	Column	Peptide	Antigen/Protein ID	Red Raw Mean	Red Background Mean	Red Foreground Mean	Red Raw Median	Red Background Median	Red Foreground Median	Flag	Annotation
1	NYGKYE Analoging	6	22	NYGKYE	protein_substitution1	234,37	48,56	185,81	249,00	45,00	204,00		
2	NYGKYE Analoging	6	24	NYGKYE	protein_substitution1	229,75	48,56	181,19	248,00	45,00	203,00		
3	NYGKYE Analoging	6	21	NYGKYE	protein_substitution1	229,38	48,56	180,82	247,00	45,00	202,00		
4	NYGKYE Analoging	6	23	NYGKYE	protein_substitution1	227,56	48,56	179,00	247,00	45,00	202,00		
5	NYGKYE Analoging	6	25	NYGKYE	protein_substitution1	223,89	48,56	175,33	250,00	45,00	205,00		
6	NYGKYE Analoging	6	18	NYGKYV		181,30	48,56	132,74	203,00	45,00	158,00		
7	NYGKYE Analoging	6	14	NYGKYQ		176,12	48,56	127,56	201,00	45,00	156,00		

Figure 26: Search for Antigens and Proteins.

7 Product Activation

The product activation requires an *internet connection*. You need to have a *serial number* obtained from the software provider or its distributors.

(1) In the *Evaluation Time Has Expired* dialog, click the *Next* button.

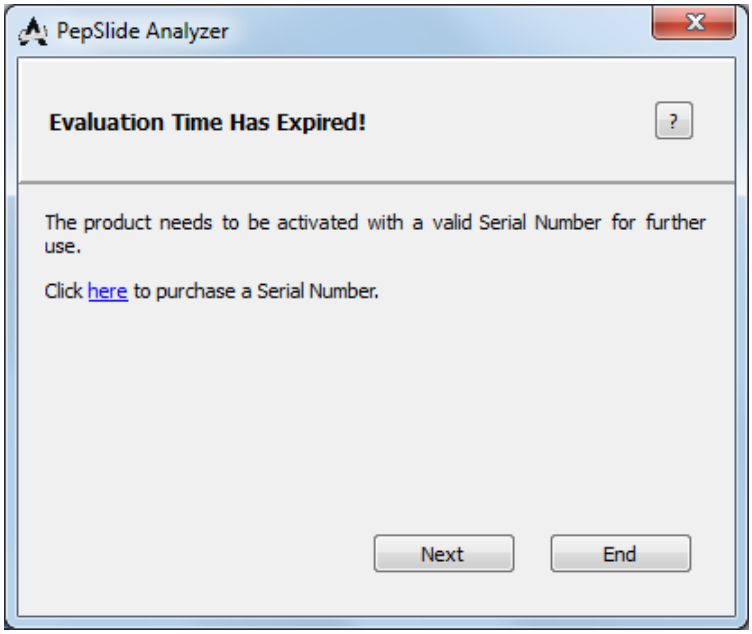


Figure 27: Starting the Product Activation.

(2) Enter the serial number and the licensee information in the *Product Activation* dialog. Click *Next*.

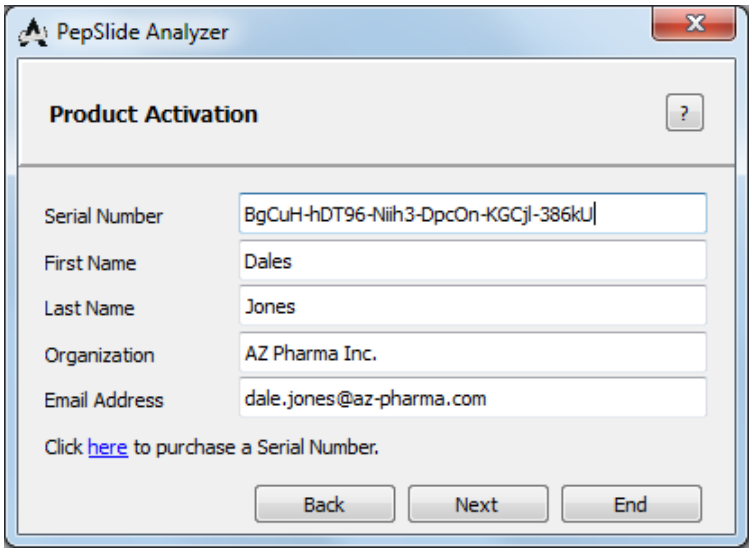


Figure 28: Entering the licensee information.

(3) If the internet connection is ready, click the *Activate* button and wait for the activation to finish.

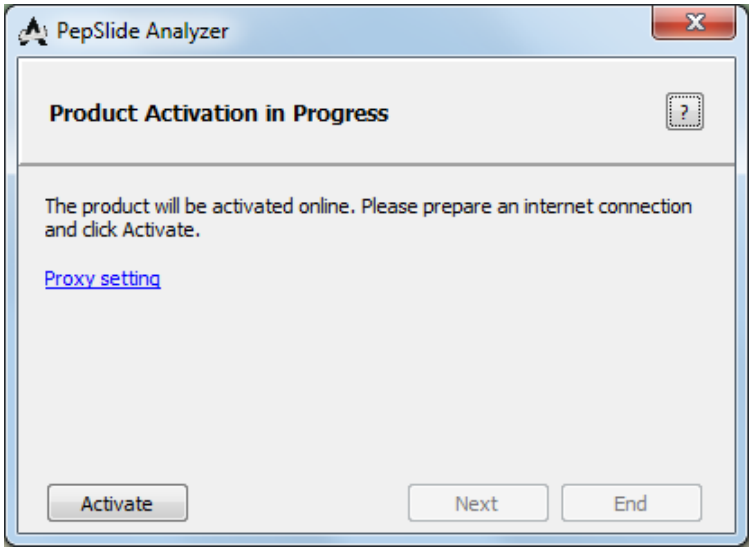


Figure 29: Product Activation in Progress.

Please check the internet connection in the case PSA could not reach the activation server. If your system uses a proxy server to connect, specify it using the *Proxy setting* link. Otherwise, contact the software provider for support.

(4) A completion message is shown when the product is successfully activated. Click *Next* to use PSA immediately or *End* to use it later.

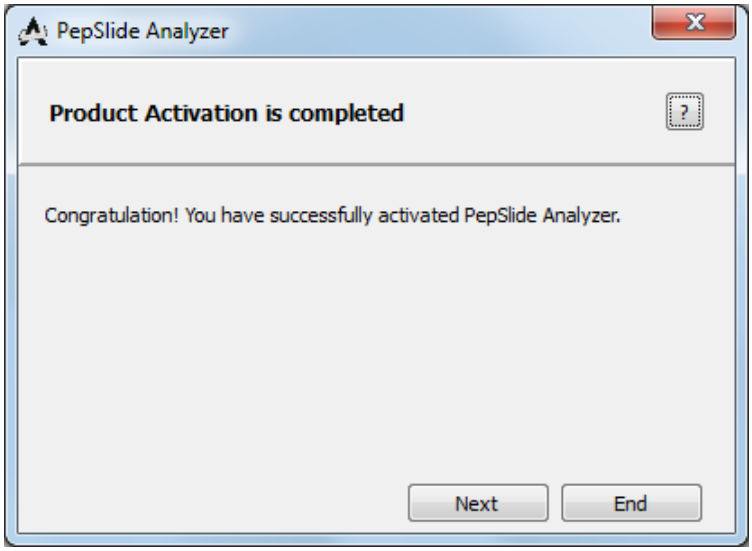


Figure 30: Completion of the Product Activation.

Please note that licenses can be reviewed, or renewed in the case of a term license, by clicking on the *Help* menu and choosing *License Information*.

8 Sample Datasets

The sample data package consists of sample datasets that can be used to get familiar with PepSlide® Analyzer software**.**

The *overlapping1*, *random1,* and *substitution1* folders contain the data of peptide arrays whose array file is of PSF format. The *overlapping3* folder also contains peptide array data, but the array file is prepared in the GAL format. You can perform data quantification (Sections 2 and Section 3) as well as analysis of peptide library (Section 4) on these datasets.

A dataset for testing data mining tools (Section 5) is stored in the *schroeder2010* folder.

Further information of the datasets is described below.

overlapping1

The peptide array consists of *overlapping* peptide libraries. The signals are recorded in a 16-bit grayscale TIFF image.

Reference to Analysis Functions: Section 4.1.

random1

The peptide array consists of *random* peptide libraries. The array's scanned image is a 24-bit color TIFF file, having the control signals in the green channel and the peptide signals in the red channel.

Reference to Analysis Functions: Section 4.2 and Section 5.4.

substitution1

The peptide array contains a *substitution* peptide library. The signals are recorded in an 8-bit grayscale TIFF image.

Reference to Analysis Functions: Section 4.3.

overlapping3

This peptide array contains an *overlapping* peptide library. Since the array file is of *GAL* format, you can try creating a peptide library for GAL blocks with this dataset.

In this folder, file overlapping3_design.txt stores the design information of the peptide library, while file PeptideMappingFile.txt contains the peptide mapping data. The signals are recorded in a 16-bit grayscale TIFF image.

Reference to Analysis Functions: Section 4.4 and Section 4.1.

schroeder2010

This folder contains a dataset compiled from this work[3]. The original GenePix Result (GPR) files can be downloaded here.

You can test the Hierarchical Clustering Analysis and Principal Component Analysis with this dataset. In this folder, file schroeder2010.csv stores the Log Ratio (549/649) values of the GPR files. The same data is stored in file schroeder2010_filtered.csv, excepted that positional controls (starting with ps_) have been filtered out.

Reference to Analysis Functions: Section 5.2 and Section 0.

antigenProteinList

This folder contains two example CSV files which can be used to test the Search for Antigens & Protein function. You can also use them a template to create your own list. The files contain the same list of proteins but are stored in two different formats:

- Tab delimited: antigenProteinList_tab.csv.
- Semicolon delimited: antigenProteinList_semicolon.csv.

References to Utilities: Section 6.1.

[3] Schröder et al. (2010): Dual-color proteomic profiling of complex samples with a microarray of 810 cancer-related antibodies, Molecular & Cellular Proteomics 9(6): 1271.

9 Customer License Agreement

Customer License Agreement for User of PepSlide® Analyzer.

PEPSLIDE ANALYZER IS THE PROPERTY OF SICASYS SOFTWARE GMBH (SICASYS). THE USE OF THIS SOFTWARE IS GOVERNED BY THE TERMS AND CONDITIONS OF THE AGREEMENT SET FORTH BELOW. THE TERM "SOFTWARE" ALSO INCLUDES RELATED DOCUMENTATION (WHETHER IN PRINT OR ELECTRONIC FORM) AND ANY UPDATES OR UPGRADES OF THE SOFTWARE PROVIDED BY SICASYS.

BY INSTALLING THE SOFTWARE, AND/OR BY USING THE SOFTWARE, YOU AGREE TO BE BOUND BY THE TERMS AND CONDITIONS OF THIS SOFTWARE LICENSE AGREEMENT.

License. SICASYS grants to you (USER) a non-exclusive, non-transferable license of use for the SOFTWARE on a number of computers stated in the license contract. A separate license is required for use on any additional computer.

Copy Restriction. The SOFTWARE may not be copied either in full or part by USER, with the exception of making copy for security or backup purpose. Copies must include all copyright and trademark notices.

Use Restrictions. This SOFTWARE is licensed to USER for internal use only. USER shall not (and shall not allow any third party to):

(i) decompile, disassemble, reverse engineer or attempt to reconstruct, identify or discover any source code, underlying ideas, underlying user interface techniques or algorithms of the SOFTWARE by any means whatever, or disclose any of the foregoing;
(ii) modify, incorporate into or with other software, or create a derivative work of any part of the SOFTWARE;
(iii) attempt to circumvent any user limits, or other license, timing or use restrictions that are built into the SOFTWARE.

Ownership of the SOFTWARE. SICASYS retains all titles of ownership, all ownership rights, and all intellectual property rights of the SOFTWARE. SICASYS reserves all rights not expressly granted to USER.

Termination. SICASYS may terminate this Agreement immediately if USER breaches any provision. Upon notice of termination by SICASYS, all rights granted to USER under this Agreement will immediately terminate, and USER shall cease using the SOFTWARE and return or destroy all copies of the SOFTWARE.

Limited Warranty and Disclaimer. USER is aware of the fact that technical errors in the program in accordance with the accompanying documentation cannot be excluded. If USER claims deviations in the program from the specification/description, USER has the right to request a fix, patch, work-around, or replacement of the SOFTWARE that does not meet such limited warranty. If a rectification of the errors is not possible or if such rectification is not successful, USER has the right to request a rescission of the contract, where USER must destroy all copies of the SOFTWARE.

EXCEPT AS EXPRESSLY SET FORTH ABOVE, NO OTHER WARRANTIES OR CONDITIONS, EITHER EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, ARE MADE BY SICASYS WITH RESPECT TO THE SOFTWARE AND THE ACCOMPANYING DOCUMENTATION, AND SICASYS EXPRESSLY DISCLAIMS ALL WARRANTIES AND CONDITIONS NOT EXPRESSLY STATED HEREIN, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY, NONINFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE. SICASYS DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET USER'S REQUIREMENTS, BE UNINTERRUPTED OR ERROR FREE, OR THAT ALL DEFECTS IN THE PROGRAM WILL BE CORRECTED. USER ASSUMES THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE SOFTWARE.

Limitation of Liability. IN NO EVENT SHALL SICASYS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL OR INCIDENTAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTIONS, LOSS OF BUSINESS INFORMATION, OR OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF SICASYS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. FURTHER, IN NO EVENT SHALL SICASYS BE LIABLE FOR ANY DIRECT DAMAGES ARISING OUT OF USER'S USE OF THE SOFTWARE. IN NO EVENT WILL SICASYS BE LIABLE TO USER FOR DAMAGES IN AN AMOUNT GREATER THAN THE FEES PAID FOR THE USE OF THE SOFTWARE.

Intellectual Property Right Infringement. If a claim alleging infringement of an intellectual property right arises concerning the SOFTWARE (including but not limited to patent, trade secret, copyright or trademark rights), SICASYS in its sole discretion may elect to defend or settle such claim, and/or terminate this Agreement and all rights to use the SOFTWARE, and require the return or destruction of the SOFTWARE, with a refund of the fees paid for use of the SOFTWARE less a reasonable allowance for use and shipping.

Miscellaneous. This Agreement is the entire agreement between USER and SICASYS with respect to the license to the SOFTWARE, and supersedes any previous oral or written communications or documents (including, if USER is obtaining an update, any agreement that may have been included with the initial version of the Software). This Agreement is governed by the laws of Germany. If any provision, or portion thereof, of this Agreement is found to be invalid or unenforceable, it will be enforced to the extent permissible and the remainder of this Agreement will remain in full force and effect. Failure to prosecute a party's rights with respect to a default hereunder will not constitute a waiver of the right to enforce rights with respect to the same or any other breach.

10 Index

A

activate · 2, 26, 27
aligning · 6
  block · 7
  spot family · 6
analysis of peptide array data · 3
analysis of peptide library · 3, 13
  GAL files · 13, 17
  overlapping peptide library · 13
  random peptide library · 15
  substitution peptide library · 15
array · 2
  array · 2
  microarray · 2
  peptide array diagram · 2

B

background control · 11
  association · 11
background correction · 10, 12
  background control · 11
  for a block · 12
  for a spot family · 11
  global background value · 11
  local method · 10, 12
background value · 8
block · 3, 4
brightness · 5, 6, 9

C

consensus motif · 13, 14
contrast · 5, 6, 9
  automatic enhancement · 5
control · 3
  background control · 3
  marker control · 3

D

data mining tools · 3, 19, 20

F

file
  *.gal · 4
  *.psa · 3
  *.psf · 4
  GAL · 4
  PepSlide® Analyzer · 3
  PepSlide® Designer · 4
flag · 8
foreground value · 8

H

hardware requirement · 1
HC · 19
Hierarchical Clustering Analysis · 19, 22
  input dataset · 20

I

Installation · 1
  Mac OS X platforms · 1
  Windows Platforms · 1
inversion · 5

K

K-Means Clustering Analysis · 19, 23

L

license · 2
  Customer License Agreement · 30
  perpetual license · 2
  term license · 2

N

noise filtering · 5

P

PCA · 19
PepSlide® · 1
  PepSlide® Analyzer · 1
  PepSlide® Designer · 1
  PSA · 1
  PSD · 1

peptide library · 3, 17
design · 17
overlapping · 13
permutation · 13
positional scanning · 13
random · 13
substitution · 13
Principal Component Analysis · 19, 20
input dataset · 20

Q

quantification · 3, 8, 9, 10, 12
change of intensity values · 9

R

raw value · 8
replicas · 9
aggregate results · 9
rotating · 6
array (spot families or blocks) · 6
images · 6

S

sample data package · 14, 15, 16, 18, 21, 22, 23, 25, 28
search for antigens or proteins · 24, 29
selecting · 7
block · 7
spot family · 7
serial number · 2, 26
SF · 3
spot · 3, 5
control spot · 3
distance between spots · 7
peptide spot · 3
size · 7
spot detection method · 10
Fixed-Spot · 10
other · 10
spot family · 3, 4
spotted region · 5, 6, 10

Z

zoom · 4